(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 4 572 223 A1**

(12) **DEMANDE DE BREVET EUROPEEN**

(43) Date de publication:
18.06.2025 Bulletin 2025/25

(21) Numéro de dépôt: **24209136.1**

(22) Date de dépôt: **28.10.2024**

(51) Classification Internationale des Brevets (IPC):
H04L 9/00 *(2022.01)*

(52) Classification Coopérative des Brevets (CPC):
H04L 9/002; H04L 2209/16

(84) Etats contractants désignés:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC ME MK MT NL NO PL PT RO RS SE SI SK SM TR
Etats d'extension désignés:
BA
Etats de validation désignés:
GE KH MA MD TN

(30) Priorité: **13.12.2023 FR 2314084**

(71) Demandeur: **Idemia France**
92400 Courbevoie (FR)

(72) Inventeurs:
- **DOTTAX, Emmanuelle**
  92400 Courbevoie (FR)
- **GIRAUD, Christophe**
  92400 Courbevoie (FR)
- **BARBU, Guillaume**
  92400 Courbevoie (FR)

(74) Mandataire: **Idemia**
2, place Samuel de Champlain
92400 Courbevoie (FR)

(54) **PROCÉDÉ POUR LE CALCUL D'UNE FONCTION DANS UN CONTEXTE BOÎTE BLEANCHE ET SYSTÈME ASSOCIÉ**

(57) Procédé pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie, le procédé étant mis en oeuvre par un système comprenant un élément sécurisé et une implémentation en boîte blanche, le procédé comprenant :
- le calcul (E100) par l'implémentation en boîte blanche, d'une donnée de traitement à partir du message d'entrée,
- le chiffrement (E200) par l'implémentation en boîte blanche, de la donnée de traitement,
- l'envoi (E300) de la donnée de traitement chiffrée à l'élément sécurisé,
- l'obtention (E400) par l'élément sécurisé d'une donnée résultat à partir de la donnée de traitement chiffrée, la donnée résultat étant l'image de la donnée de traitement par une fonction intermédiaire,
- le calcul (E500) du message de sortie à partir de la donnée résultat.

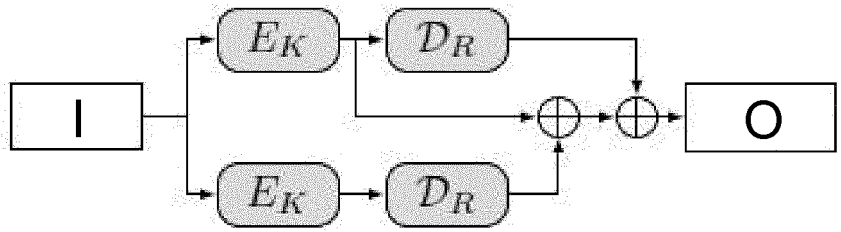

Figure 3

EP 4 572 223 A1

Description

[0001] La présente invention concerne un procédé pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie, ainsi qu'un système associé.

[0002] Une fonction sécurisée, par exemple une fonction cryptographique, utilise de façon conventionnelle des données ayant vocation à rester secrètes, par exemple une clé cryptographique sous-jacente. Lorsqu'une fonction sécurisée est mise en oeuvre au moyen d'un logiciel exécuté dans un environnement non sécurisé, des mesures particulières doivent être prises pour éviter qu'un attaquant puisse avoir accès aux données secrètes.

[0003] La recherche de techniques permettant de sécuriser la mise en oeuvre d'une fonction dans un environnement non sécurisé est connue sous le nom de cryptographie en boîte blanche (en anglais : "white box cryptography").

[0004] Les implémentations logicielles qui permettent de sécuriser la mise en oeuvre d'une fonction dans un environnement non sécurisé sont connues sous le nom d'implémentations en boite blanche. L'article "White Box Cryptography and an AES implementation", de S. Chow et al. in, Post-Proceedings of the 9th Annual Workshop on Selected Areas in Cryptography (SAC'02), 15-16 août 2002, propose par exemple une technique pour produire des algorithmes de type AES adaptés chacun à une clé cryptographique particulière.

[0005] Pour les solutions généralement proposées dans ce cadre, la fonction sécurisée est décomposée en une série de traitements élémentaires et des tables de correspondance (en anglais : "Look-Up Tables") associées respectivement à ces traitements élémentaires sont utilisées pour manipuler des données masquées.

[0006] Toutefois, une implémentation en boite blanche peut ne pas garantir un niveau de sécurité suffisant. Certaines fonctions sécurisées sont destinées à être exécutées dans des environnements agréés. Par exemple, une fonction sécurisée permettant l'accès à un véhicule, est censée n'être exécutée que sous l'environnement d'un utilisateur agréé.

[0007] Pour deviner les données secrètes de la fonction sécurisée, une attaque dite « par clonage » (« code lifting » en anglais) consiste à copier l'implémentation en boite blanche de la fonction sécurisée vers un environnement non sécurisé sous le contrôle complet d'un attaquant.

[0008] Dans un tel environnement, l'attaquant peut exécuter de très nombreuses itérations de la fonction sécurisée et/ou utiliser des outils, typiquement un outil de débogage, pour exécuter la fonction sécurisée par étape. En outre, avec une telle attaque l'attaquant peut se substituer à l'utilisateur agréé et substituer un environnement sous son contrôle à l'environnement agréé, par exemple pour accéder à un véhicule dudit utilisateur.

[0009] Pour contrer ce type d'attaque, une première solution décrite dans la demande de brevet US2019312718 consiste en une implémentation en boite blanche qui utilise sous forme déchiffrée une clé de cryptage codée reçue d'un environnement d'exécution de confiance.

[0010] Cette solution présente toutefois comme inconvénient d'être insuffisamment sécurisée ou de nécessiter une connexion avec un serveur pour recevoir une clé.

[0011] Une deuxième solution connue consiste en une implémentation en boite blanche qui envoie une donnée à un élément sécurisé. L'élément sécurisé calcule un résultat de l'application d'une fonction sur ladite donnée et l'implémentation en boite blanche vérifie ensuite que le résultat calculé par l'élément sécurisé correspond à l'application de la fonction sur ladite donnée (par exemple en calculant un autre résultat de l'application de la fonction sur ladite donnée ou de l'application de l'inverse de ladite fonction sur le résultat).

[0012] Cette deuxième solution a pour inconvénient d'être aussi insuffisamment sécurisée et d'être couteuse en ressources, typiquement en temps de calcul et en taille mémoire, pour l'environnement non sécurisé mettant en oeuvre la solution.

[0013] Une troisième solution consiste à utiliser un élément sécurisé pour mettre à jour l'implémentation en boite blanche et modifier ainsi le comportement de ladite implémentation en boite blanche. Malheureusement la mise en oeuvre de cette solution nécessite le remplacement d'une table de correspondance de l'implémentation en boite blanche, ce qui est coûteux en ressources pour le système mettant en oeuvre cette solution, notamment pour l'élément sécurisé.

[0014] Pour remédier à ces inconvénients, la présente invention propose selon un premier aspect, un procédé pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie, le procédé étant mis en oeuvre par un système comprenant un élément sécurisé et une implémentation en boîte blanche dans un environnement d'exécution non sécurisé, le procédé étant caractérisé en ce qu'il comprend les étapes suivantes :

- calcul d'une donnée de traitement à partir du message d'entrée par l'implémentation en boîte blanche,
- chiffrement de la donnée de traitement en utilisant une clé de chiffrement, par l'implémentation en boîte blanche,
- envoi de la donnée de traitement chiffrée à l'élément sécurisé,
- obtention par l'élément sécurisé d'une donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, la donnée résultat étant l'image de la donnée de traitement par une fonction intermédiaire différente de la fonction identité,
- calcul du message de sortie à partir de la donnée résultat.

[0015] D'autres caractéristiques avantageuses et non limitatives du procédé conforme à l'invention, prises individuellement ou selon toutes les combinaisons techniquement possibles, sont les suivantes :

- le message de sortie comprend la donnée résultat ;
- le procédé comprenant en outre une étape de calcul d'au moins une autre donnée de traitement par l'implémentation en boîte blanche, et le message de sortie comprend en outre l'autre donnée de traitement ;

  - - l'implémentation en boite blanche a une première partie et une deuxième partie, le calcul d'une donnée de traitement à partir du message d'entrée et le chiffrement de la donnée de traitement en utilisant une clé de chiffrement, sont mis en oeuvre par la première partie de l'implémentation en boite blanche, le calcul du message de sortie à partir de la donnée résultat est mis en oeuvre par la deuxième partie de l'implémentation en boite blanche, et le procédé comprend en outre une étape d'envoi de la donnée résultat à la deuxième partie de l'implémentation en boite blanche ;

- le procédé comprend en outre le calcul d'au moins une autre donnée de traitement par la première partie de l'implémentation en boîte blanche, et l'utilisation de l'au moins une autre donnée de traitement par la deuxième partie de l'implémentation en boîte blanche pour le calcul du message de sortie à partir de la donnée résultat ;
- l'obtention par l'élément sécurisé de la donnée résultat, est par déchiffrement fonctionnel de la donnée de traitement chiffrée en utilisant une clé de déchiffrement fonctionnel pour la fonction intermédiaire, la clé de déchiffrement étant ladite clé de déchiffrement fonctionnel pour la fonction intermédiaire ;
- la clé de chiffrement est une clé publique d'un algorithme RSA et la clé de déchiffrement fonctionnel est obtenue à partir d'une clé privée dudit algorithme RSA, la clé privé étant associée à la clé publique, l'algorithme RSA ayant un module de chiffrement déterminé et la fonction intermédiaire étant une exponentiation modulaire élevant la donnée de traitement à un exposant déterminé modulo le module de chiffrement déterminé ;
- la clé de chiffrement comprend un exposant de chiffrement et la clé de déchiffrement fonctionnel comprend un inverse modulaire de l'exposant de chiffrement et l'exposant déterminé, ou le résultat du produit de l'exposant de chiffrement par l'exposant déterminé ;
- la clé de chiffrement n'intervient que dans l'étape de chiffrement de la donnée de traitement par l'implémentation en boîte blanche, et la clé de déchiffrement n'intervient que dans l'étape d'obtention d'une donnée résultat par l'élément sécurisé ;
- la fonction sécurisée se compose d'une suite d'opérations et la fonction intermédiaire est une partie de ladite suite d'opérations ;
- la fonction sécurisée est une fonction cryptographique mettant en correspondance le message d'entrée avec le message de sortie à l'aide d'une clé cryptographique prédéterminée ;
- la clé cryptographique prédéterminée est une clé distincte de la clé de chiffrement et de la clé de déchiffrement.

[0016] Au moins une partie des procédés selon l'invention peut être mise en oeuvre par ordinateur. En conséquence, la présente invention peut prendre la forme d'un mode de réalisation combinant des aspects logiciels (comportant les microprogrammes, les logiciels résidents, les microcodes, etc.) et matériels qui peuvent tous être globalement appelés ici "composant".

[0017] Selon un deuxième aspect, l'invention propose un système pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie, le système étant caractérisé en ce qu'il comprend :

- un premier composant comprenant tout ou partie d'une implémentation en boite blanche dans un environnement d'exécution non sécurisé, ladite toute ou partie de l'implémentation en boite blanche étant configurée pour calculer une donnée de traitement à partir du message d'entrée et chiffrer la donnée de traitement en utilisant une clé de chiffrement,
- un élément sécurisé configuré pour recevoir la donnée de traitement chiffrée et obtenir une donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, la donnée résultat étant l'image de la donnée de traitement par une fonction intermédiaire différente de la fonction identité,
- un deuxième composant configuré pour recevoir la donnée résultat et calculer le message de sortie à partir de la donnée résultat.

[0018] D'autres caractéristiques avantageuses et non limitatives du système conforme à l'invention, prises individuellement ou selon toutes les combinaisons techniquement possibles, sont les suivantes :

- le message de sortie comprend la donnée résultat ;
- l'implémentation en boite blanche a une première partie et une deuxième partie, ladite toute ou partie de l'implémentation en boite blanche du premier composant est la première partie de l'implémentation en boite blanche, le deuxième composant comprend la deuxième partie de l'implémentation en boite blanche dans l' environnement d'exécution non sécurisé, la deuxième partie de l'implémentation

en boite blanche étant configurée pour recevoir la donnée résultat et calculer le message de sortie à partir de la donnée résultat.

[0019] Ce système peut être configuré pour la mise en oeuvre de chacune des possibilités de réalisation envisagées pour le procédé tel que défini précédemment.

[0020] Bien entendu, les différentes caractéristiques, variantes et formes de réalisation de l'invention peuvent être associées les unes avec les autres selon diverses combinaisons dans la mesure où elles ne sont pas incompatibles ou exclusives les unes des autres.

[0021] D'autres caractéristiques et avantages de la présente invention ressortiront de la description faite ci-dessous, en référence aux figures annexées qui en illustrent des exemples de réalisation dépourvus de tout caractère limitatif.

[0022] Sur les figures :

Figure 1 représente schématiquement un mode de réalisation préféré d'un système selon l'invention, notamment pour la mise en oeuvre d'un procédé selon l'invention ;

Figure 2 illustre sous forme de logigramme les étapes principales d'un procédé pour effectuer une fonction sécurisée selon l'invention ;

Figure 3 illustre un exemple de fonction sécurisée effectuée par un procédé ou un système selon l'invention.

[0023] Sauf indications contraires, les éléments communs ou analogues à plusieurs figures portent les mêmes signes de référence et présentent des caractéristiques identiques ou analogues, de sorte que ces éléments communs ne sont généralement pas à nouveau décrits par souci de simplicité.

[0024] Dans le cadre de la présente description, des qualificatifs « premier », « deuxième », « troisième » et « quatrième » ne sont qu'à titre indicatif pour distinguer des éléments qu'ils qualifient, mais n'impliquent pas d'ordre entre eux.

[0025] La figure 1 représente schématiquement un mode de réalisation préféré d'un système 1 selon l'invention.

[0026] Le système 1 comprend un environnement d'exécution non sécurisé 2 et un élément sécurisé 3.

[0027] Le système 1 est adapté pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie.

[0028] L'environnement d'exécution non sécurisé 2 comprend un moyen de traitement de données 20 de type processeur, un moyen de stockage de données 21, une mémoire vive 22, une première interface de communication 23 et une deuxième interface de communication 24.

[0029] Le moyen de stockage de données 21 et la mémoire vive 22 de l'environnement d'exécution non sécurisé 2 sont chacun liés au moyen de traitement de données 20 dudit environnement d'exécution non sécurisé 2, de sorte que le moyen de traitement de données 20 peut lire ou écrire des données dans le moyen de stockage de données 21 et/ou la mémoire vive 22.

[0030] Le moyen de stockage de données 21 mémorise des instructions de programme d'ordinateur, dont certaines sont conçues pour mettre en oeuvre des étapes d'un procédé pour effectuer une fonction sécurisée tel que décrit en référence à la figure 2 lorsque ces instructions sont exécutées par le moyen de traitement de données 20.

[0031] Le moyen de stockage de données 21 est par exemple en pratique un disque dur ou une mémoire non-volatile, éventuellement réinscriptible, par exemple de type EEPROM (pour "Electrically Erasable and Programmable Read-Only Memory" selon l'appellation anglo-saxonne couramment utilisée).

[0032] En outre, le moyen de stockage de données 21 et la mémoire vive 22 peuvent mémoriser certains au moins des éléments (notamment la donnée de traitement chiffrée et la donnée résultat telles que décrites ci-après en référence à la figure 2) manipulés lors des différents traitements effectués au cours du procédé décrit ci-après.

[0033] On appelle mémoire dans la suite de la description, l'un quelconque parmi le moyen de stockage de données 21 et la mémoire vive 22.

[0034] L'environnement d'exécution non sécurisé 2 comporte également plusieurs composants non représentés.

[0035] Typiquement, l'environnement d'exécution non sécurisé 2 comprend un premier composant et un deuxième composant.

[0036] Ces composants peuvent en pratique être réalisés par une combinaison d'éléments matériels et d'éléments logiciels.

[0037] Chaque composant est configuré pour réaliser une étape d'un procédé conforme à l'invention, et possède donc une fonctionnalité décrite dans le procédé conforme à l'invention et exposé ci-après. Le premier composant comprend tout ou partie d'une implémentation en boite blanche.

[0038] Le système 1 comprend donc une implémentation en boite blanche dans l'environnement d'exécution non sécurisé 2. Le système 1, typiquement l'environnement d'exécution non sécurisé 2, mémorise l'implémentation en boite blanche.

[0039] Selon un premier exemple de réalisation des composants, l'implémentation en boite blanche a une première partie et une deuxième partie, ladite toute ou partie de l'implémentation en boite blanche du premier composant est la première partie de l'implémentation en boite blanche, et le deuxième composant comprend la deuxième partie de l'implémentation en boite blanche.

[0040] Selon un deuxième exemple de réalisation des composants, le premier composant comprend tout ou partie de l'implémentation en boite blanche, et le deuxième composant ne comprend pas tout ou partie de

l'implémentation en boite blanche.

[0041] Pour chaque composant, l'environnement d'exécution non sécurisé 2 mémorise par exemple des instructions de logiciel (également nommées instructions de programme d'ordinateur) exécutables par le moyen de traitement de 20 afin d'utiliser un élément matériel (par exemple une mémoire) et de mettre ainsi en oeuvre la fonctionnalité offerte par le composant.

[0042] Selon une possibilité de réalisation, les instructions de programme d'ordinateur mémorisées dans le moyen de stockage de donnée 21 ont été reçues (par exemple d'un ordinateur distant via la deuxième interface de communication 24) lors d'une phase de fonctionnement de l'environnement d'exécution non sécurisé 2, antérieure au procédé décrit en référence à la figure 2.

[0043] Le moyen de traitement de données 20 est donc configuré pour mettre en oeuvre certaines étapes du procédé pour effectuer une fonction sécurisée, qui sera décrit plus loin.

[0044] Le moyen de traitement de données 20 peut avoir n'importe quelle structure. Le moyen de traitement de données 20 comprend un ou plusieurs coeurs, chaque coeur étant configuré pour exécuter des instructions de code d'un programme de manière à mettre en oeuvre les étapes précitées.

[0045] La première interface de communication 23 est reliée au moyen de traitement de données 20 de manière à permettre à l'environnement d'exécution non sécurisé 2 de communiquer avec l'élément sécurisé 3 via une autre interface de communication 33 de l'élément sécurisé 3.

[0046] La première interface de communication est de type quelconque. Elle est par exemple une interface filaire moyennant un protocole de communication quelconque, par exemple de type OPC (pour « Open Platform Communications » en terminologie anglo-saxonne), ou selon la norme ISO/IEC 7816 dans une de ses versions déjà publiées.

[0047] La première interface de communication 23 permet à l'environnement d'exécution non sécurisé 2 d'envoyer des données à l'élément sécurisé 3, par exemple une donnée de traitement chiffrée telle que décrite en référence à la figure 2, et/ou de recevoir des données en provenance de l'élément sécurisé 3, par exemple une donnée résultat telle que décrite en référence à la figure 2.

[0048] La deuxième interface de communication 24 est reliée au moyen de traitement de données 20 de manière à permettre au moyen de traitement de données 20 de recevoir un message d'entrée en provenance d'un dispositif électronique non représenté et/ou d'envoyer un message de sortie audit dispositif électronique non représenté.

[0049] La deuxième interface de communication 24 est de type quelconque. Elle est par exemple filaire (Ethernet) ou de type radio sans fil moyennant un protocole de communication quelconque (Wi-Fi, Bluetooth, NFC, etc.).

[0050] L'élément sécurisé 3 comprend un autre moyen de traitement de données 30 de type processeur, un autre moyen de stockage de données 31, une autre mémoire vive 32 et l'autre interface de communication 33.

[0051] L'autre moyen de stockage de données 31 et l'autre mémoire vive 32 de l'élément sécurisé 3 sont chacun liés à l'autre moyen de traitement de données 30 dudit élément sécurisé 3, de sorte que l'autre moyen de traitement de données 30 peut lire ou écrire des données dans l'autre moyen de stockage de données 31 et/ou l'autre mémoire vive 32.

[0052] L'autre moyen de stockage de données 31 mémorise des instructions de programme d'ordinateur, dont certaines sont conçues pour mettre en oeuvre des étapes d'un procédé pour effectuer une fonction sécurisée tel que décrit en référence à la figure 2 lorsque ces instructions sont exécutées par l'autre moyen de traitement de données 30.

[0053] L'autre moyen de stockage de données 31 est par exemple en pratique une mémoire non-volatile, éventuellement réinscriptible, par exemple de type EEPROM (pour "Electrically Erasable and Programmable Read-Only Memory" selon l'appellation anglo-saxonne couramment utilisée).

[0054] En outre, l'autre moyen de stockage de données 31 et l'autre mémoire vive 32 peuvent mémoriser certains au moins des éléments (notamment la donnée de traitement chiffrée et la donnée résultat telles que décrites ci-après en référence à la figure 2) manipulés lors des différents traitements effectués au cours du procédé décrit ci-après.

[0055] On appelle autre mémoire dans la suite de la description, l'un quelconque parmi l'autre moyen de stockage de données 31 et l'autre mémoire vive 32.

[0056] L'élément sécurisé 3 mémorise par exemple des instructions de logiciel exécutables par l'autre moyen de traitement de 30 afin d'utiliser un élément matériel (par exemple une autre mémoire) et de mettre ainsi en oeuvre une, ou plusieurs, étape des procédés conformes à l'invention, et donc une, ou plusieurs, fonctionnalité décrite dans le procédé conforme à l'invention et exposé ci-après. Selon une possibilité de réalisation, les instructions de programme d'ordinateur mémorisées dans l'autre moyen de stockage de donnée 31 ont été reçues (par exemple d'un autre ordinateur distant, ou de l'environnement d'exécution non sécurisé 2, via l'autre interface de communication 33,) lors d'une phase de fonctionnement de l'élément sécurisé 3 antérieure au procédé décrit en référence à la figure 2.

[0057] L'autre moyen de traitement de données 30 est donc configuré pour mettre en oeuvre certaines étapes du procédé pour effectuer une fonction sécurisée, qui sera décrit plus loin.

[0058] L'autre moyen de traitement de données 30 peut avoir n'importe quelle structure. L'autre moyen de traitement de données 30 comprend un ou plusieurs coeurs, chaque coeur étant configuré pour exécuter

des instructions de code d'un programme de manière à mettre en oeuvre les étapes précitées.

[0059] L'autre interface de communication 33 est reliée à l'autre moyen de traitement de données 30 de manière à permettre à l'élément sécurisé 3 de communiquer avec l'environnement d'exécution non sécurisé 2 via la première interface de communication 23 de l'environnement d'exécution non sécurisé 2.

[0060] L'autre interface de communication 33 est du même type que la première interface de communication 23. Elle est par exemple une interface de filaire moyennant un protocole de communication, par exemple de type OPC (pour « Open Platform Communications » en terminologie anglo-saxonne), ou selon la norme ISO/IEC 7816 dans une de ses versions déjà publiées.

[0061] L'autre interface de communication 33 permet à l'élément sécurisé 3 d'envoyer des données à l'environnement d'exécution non sécurisé 2, par exemple une donnée résultat telle que décrite en référence à la figure 2, et/ou de recevoir des données en provenance de l'environnement d'exécution non sécurisé 2, par exemple une donnée de traitement chiffrée telle que décrite en référence à la figure 2.

[0062] Selon un premier exemple, l'environnement non sécurisé 2 est un terminal de communication, un ordinateur personnel, une tablette ou un serveur, et l'élément sécurisé 3 est une puce intégrée dans une carte à puce, telle qu'une carte d'identité, une carte bancaire ou une carte à circuit intégré universelle (également connue sous le nom de carte UICC pour « Universal Integrated Circuit Card » en terminologie anglo-saxonne) telle qu'une carte d'abonné à un réseau cellulaire, typiquement une carte SIM.

[0063] Selon un deuxième exemple, le système 1 est un terminal de communication, un ordinateur personnel, une tablette ou un serveur, et l'élément sécurisé 3 est un microcontrôleur sécurisé ou un environnement d'exécution de confiance (« Trusted Execution Environment » en terminologie Anglosaxonne, aussi dénommé par l'acronyme TEE) qui est intégré audit terminal de communication, ordinateur personnel, tablette ou serveur.

[0064] La figure 2 illustre sous forme de logigramme les étapes principales d'un procédé pour effectuer une fonction sécurisée selon l'invention.

[0065] Ce procédé est mis en oeuvre par le système 1.

[0066] La fonction sécurisée met en correspondance un message d'entrée avec un message de sortie.

[0067] La fonction sécurisée peut être une fonction cryptographique mettant en correspondance le message d'entrée avec le message de sortie à l'aide d'une clé cryptographique prédéterminée.

[0068] La fonction cryptographique comprend par exemple une fonction de chiffrement, une fonction de déchiffrement, une fonction de signature ou une fonction de vérification de signature avec la clé cryptographique prédéterminée. La clé cryptographique prédéterminée est alors de préférence une clé distincte de la clé de chiffrement et de la clé de déchiffrement décrites ci-après.

[0069] Selon une étape de calcul d'une donnée de traitement (étape E100), une donnée de traitement est calculée à partir du message d'entrée par l'implémentation en boîte blanche.

[0070] Selon une possibilité, la donnée de traitement peut être le message d'entrée ou une première partie du message d'entrée.

[0071] Selon une autre possibilité, la donnée de traitement peut être le résultat de l'application d'une première autre fonction au message d'entrée ou à une première partie du message d'entrée.

[0072] Le procédé peut comprendre alors une étape de calcul d'au moins une autre donnée de traitement (étape E110) pendant laquelle au moins une autre donnée de traitement est calculée par l'implémentation en boîte blanche.

[0073] Selon une première possibilité, l'autre donnée de traitement peut être le message d'entrée ou une deuxième partie du message d'entrée.

[0074] Selon une deuxième possibilité, l'autre donnée de traitement peut être le résultat de l'application d'une deuxième autre fonction au message d'entrée, à une deuxième partie du message d'entrée, à la donnée de traitement ou à une première partie de la donnée de traitement.

[0075] Cette étape de calcul d'au moins une autre donnée de traitement est optionnelle et peut être omise. Le procédé comprend alors une étape de chiffrement (étape E200) pendant laquelle l'implémentation en boîte blanche chiffre la donnée de traitement en utilisant une clé de chiffrement.

[0076] Typiquement, le chiffrement est selon un algorithme cryptographique asymétrique, par exemple RSA ou à base de courbes elliptiques, ou un algorithme cryptographique symétrique, par exemple DES, 3DES ouAES.

[0077] Quand le chiffrement est selon un algorithme cryptographique asymétrique, la clé de chiffrement est une clé publique au sens dudit algorithme cryptographique asymétrique.

[0078] Le procédé comprend alors une étape (étape E300) d'envoi de la donnée de traitement chiffrée à l'élément sécurisé 3 du système 1, la donnée de traitement chiffrée étant le résultat du chiffrement mis en oeuvre par l'implémentation en boîte blanche pendant l'étape de chiffrement (étape E200). Typiquement, l'environnement d'exécution non sécurisé 2 du système 1 envoie la donnée de traitement chiffrée à l'élément sécurisé 3 via la première interface de communication 23, et l'élément sécurisé 3 reçoit la donnée de traitement chiffrée via l'autre interface de communication 33.

[0079] Le procédé comprend alors une étape d'obtention (étape E400) pendant laquelle l'élément sécurisé 3 obtient une donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, la donnée résultat étant l'image de la donnée de traitement, c'est-à-dire de la donnée de

traitement non chiffrée, par une fonction intermédiaire différente de la fonction identité.

[0080] Typiquement, quand le chiffrement de l'étape de chiffrement (étape E200) est selon un algorithme cryptographique symétrique, par exemple DES, 3DES ou AES, l'élément sécurisé 3 obtient la donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, en déchiffrant la donnée de traitement chiffrée selon ledit algorithme cryptographique symétrique avec ladite clé de déchiffrement puis en appliquant la fonction intermédiaire au résultat du déchiffrement.

[0081] Quand le chiffrement de l'étape de chiffrement (étape E200) est selon un algorithme cryptographique asymétrique, par exemple RSA ou à base de courbes elliptiques, l'élément sécurisé 3 peut obtenir la donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, en déchiffrant la donnée de traitement chiffrée selon ledit algorithme cryptographique asymétrique avec ladite clé de déchiffrement puis en appliquant la fonction intermédiaire au résultat du déchiffrement. La clé de déchiffrement est alors une clé privée au sens dudit algorithme cryptographique asymétrique.

[0082] De préférence, l'algorithme cryptographique asymétrique est un algorithme de chiffrement fonctionnel.

[0083] Dans ce cas, l'obtention par l'élément sécurisé 3 de la donnée résultat, est par déchiffrement fonctionnel de la donnée de traitement chiffrée en utilisant une clé de déchiffrement fonctionnel pour la fonction intermédiaire, la clé de déchiffrement étant ladite clé de déchiffrement fonctionnel pour la fonction intermédiaire.

[0084] Le procédé est ainsi davantage sécurisé. La fonction intermédiaire est dissimulée dans la clé de déchiffrement fonctionnelle, ce qui renforce sa confidentialité.

[0085] Le procédé permet en outre de limiter les temps de calcul et l'espace mémoire consommé pour l'élément sécurisé, la mise en oeuvre du déchiffrement appliquant la fonction intermédiaire à la donnée de traitement.

[0086] Des exemples d'algorithmes de chiffrement fonctionnel sont décrits dans le document « Simple Functional Encryption Schemes for Inner Products», Michel Abdalla, Florian Bourse, Angelo De Caro, and David Pointcheval, DOI : 10.1007/978-3-662-46447-2_33.

[0087] L'algorithme de la section 3 de ce document, nommée « Inner-Product from DDH », permet ainsi l'obtention d'une donnée intermédiaire ayant pour valeur un générateur *g* élevé à une puissance égale au produit de la donnée de traitement (x dans le document cité) par une autre donnée prédéterminée (y dans le document cité), le générateur *g* appartenant à un groupe d'ordre p avec p un nombre premier. La donnée résultat peut être la donnée intermédiaire, la fonction intermédiaire étant alors l'élévation du générateur *g* à une puissance égale au produit de la donnée de traitement (x dans le document cité) par une autre donnée prédéterminée (y dans le document cité). Selon une autre possibilité, la donnée résultat peut être obtenue par l'élément sécurisé 3 à partir de la donnée intermédiaire, par exemple par logarithme discret de la donnée intermédiaire, la donnée résultat ayant alors pour valeur le résultat du produit de la donnée de traitement par l'autre donnée prédéterminée et la fonction intermédiaire étant le produit de la donnée de traitement par l'autre donnée prédéterminée. On notera que la clé de déchiffrement et la clé de chiffrement sont respectivement sk_y et mpk dans la section 3 du document cité.

[0088] De manière préférée, l'algorithme de chiffrement fonctionnel est un algorithme RSA ayant un module de chiffrement déterminé *N* et la fonction intermédiaire est une exponentiation modulaire élevant la donnée de traitement à un exposant déterminé modulo le module de chiffrement déterminé.

[0089] Ainsi, la clé de chiffrement est une clé publique d'un algorithme RSA et la clé de déchiffrement fonctionnel est obtenue à partir d'une clé privée dudit algorithme RSA, la clé privée étant associée à la clé publique, l'algorithme RSA ayant un module de chiffrement déterminé et la fonction intermédiaire étant une exponentiation modulaire élevant la donnée de traitement à un exposant déterminé modulo le module de chiffrement déterminé.

[0090] Le procédé permet ainsi une mise en oeuvre simple et peu coûteuse en ressource de la fonction intermédiaire par déchiffrement fonctionnel.

[0091] Typiquement, la clé de chiffrement comprend un exposant de chiffrement e et la clé de déchiffrement fonctionnel comprend un inverse modulaire d de l'exposant de chiffrement e et l'exposant déterminé *a*, ou le résultat du produit de l'exposant de chiffrement par l'exposant déterminé, c'est-à-dire $a \times d$.

[0092] Ainsi, pendant l'étape de chiffrement (étape E200), l'implémentation en boîte blanche peut chiffrer la donnée de traitement comme suit : $u = x^e \ mod \ N$, avec *x* la donnée de traitement et *u* la donnée de traitement chiffrée.

[0093] Pendant l'étape d'obtention (étape E400) l'élément sécurisé 3 peut obtenir la donnée résultat comme suit : $z = u^{a \times d} \ mod \ N$, avec z la donnée résultat.

[0094] La donnée résultat z a donc pour valeur $x^a \ mod \ N$. Autrement dit, la fonction intermédiaire est une exponentiation modulaire élevant la donnée de traitement *x* à l'exposant déterminé *a* modulo le module de chiffrement déterminé *N*.

[0095] L'exposant déterminé *a* est un entier, par exemple 2.

[0096] De manière préférée, la fonction intermédiaire ne met pas en oeuvre tout ou partie de la clé cryptographique.

[0097] On a ainsi plus de liberté pour choisir la fonction sécurisée et la clé cryptographique, déployer la clé cryptographique, et gérer les accès associés.

[0098] Le procédé comprend alors une étape de calcul du message de sortie (étape E500), pendant laquelle le

système 1, calcule le message de sortie à partir de la donnée résultat.

[0099] Selon mode particulier de réalisation du procédé, typiquement quand le premier composant et le deuxième composant du système 1 sont réalisés selon le premier exemple de réalisation des composants décrit en référence à la figure 1, l'implémentation en boite blanche a une première partie et une deuxième partie, le calcul (étape E100) d'une donnée de traitement à partir du message d'entrée et le chiffrement (étape E200) de la donnée de traitement en utilisant une clé de chiffrement, sont mis en oeuvre par la première partie de l'implémentation en boite blanche, et le calcul (étape E500) du message de sortie à partir de la donnée résultat est mis en oeuvre par la deuxième partie de l'implémentation en boite blanche.

[0100] Le procédé comprend alors, typiquement entre l'étape d'obtention (étape E400) et l'étape de calcul du message de sortie (étape E500), une étape (non représentée) d'envoi de la donnée résultat à la deuxième partie de l'implémentation en boîte blanche.

[0101] Typiquement, l'élément sécurisé 3 du système 1 envoie la donnée résultat à la deuxième partie de l'implémentation en boîte blanche, en envoyant la donnée résultat à l'environnement d'exécution non sécurisé 2 du système 1 via l'autre interface de communication 33, l'environnement d'exécution non sécurisé, et la deuxième partie de l'implémentation en boîte blanche, recevant la donnée résultat via la première interface de communication 23.

[0102] Selon un autre mode particulier de réalisation du procédé, typiquement quand le premier composant et le deuxième composant du système 1 sont réalisés selon le deuxième exemple de réalisation des composants décrit en référence à la figure 1, le calcul (étape E500) du message de sortie à partir de la donnée résultat n'est pas mis en oeuvre par tout ou partie de l'implémentation en boite blanche. Le procédé peut comprendre alors, typiquement entre l'étape d'obtention (étape E400) et l'étape de calcul du message de sortie (étape E500), une étape (non représentée) d'envoi de la donnée résultat à l'environnement non sécurisé 2.

[0103] Typiquement, l'élément sécurisé 3 du système 1 peut envoyer la donnée résultat à l'environnement non sécurisé 2 du système 1 via l'autre interface de communication 33, l'environnement d'exécution non sécurisé 2 recevant la donnée résultat via la première interface de communication 23.

[0104] De préférence, quand le procédé est selon l'autre mode particulier de réalisation, le message de sortie comprend la donnée résultat. Le message de sortie peut être la donnée résultat.

[0105] Le procédé, notamment selon le mode particulier de réalisation ou l'autre mode particulier de réalisation décrits ci-avant, permet de limiter les temps de calcul et l'espace mémoire consommé, notamment par l'implémentation boite blanche.

[0106] Le message de sortie est calculé à partir de la donnée résultat.

[0107] L'étape d'obtention (étape E400) mise en oeuvre par l'élément sécurisé 3 contribue aux calculs de la fonction sécurisée en obtenant la donnée résultat à partir de la donnée de traitement.

[0108] Le système 1, notamment la deuxième partie de l'implémentation en boite blanche, n'a pas besoin de vérifier que la donnée résultat correspond au résultat de l'application de la fonction intermédiaire à la donnée de traitement. Si la donnée résultat ne correspond pas au résultat de l'application de la fonction intermédiaire à la donnée de traitement, le message de sortie obtenu sera erroné. En effet, le message de sortie ne sera pas l'image du message d'entrée par la fonction sécurisée.

[0109] Le calcul de l'étape de calcul d'une donnée de traitement (étape E100), la fonction intermédiaire et le calcul de l'étape de calcul du message de sortie (E500) sont donc choisis de sorte à mettre en oeuvre la fonction sécurisée.

[0110] Le procédé est en outre particulièrement sécurisé.

[0111] Un attaquant observant les échanges de données entre l'implémentation en boite blanche (et/ou l'environnement non sécurisé 2) et l'élément sécurisé 3 n'accède pas à la donnée de traitement et ne peut pas en déduire la fonction intermédiaire, car la donnée de traitement est chiffrée.

[0112] Au sein du système 1, la clé de chiffrement est de préférence uniquement intégrée dans l'implémentation en boite blanche et n'est pas stockée en tant que telle dans la mémoire du système 1.

[0113] Quand le procédé est selon le mode particulier de réalisation décrit ci-avant, la clé de chiffrement est de préférence uniquement intégrée dans la première partie de l'implémentation en boite blanche et n'est pas stockée en tant que telle dans la mémoire du système 1.

[0114] Au sein du système 1, la clé de déchiffrement est de préférence stockée uniquement dans l'élément sécurisé 3 et n'est pas intégrée dans l'implémentation en boite blanche.

[0115] De manière également préférée, la clé de chiffrement n'intervient que dans l'étape de chiffrement de la donnée de traitement (étape E200) par l'implémentation en boîte blanche, et la clé de déchiffrement n'intervient que dans l'étape d'obtention (étape E400) d'une donnée résultat par l'élément sécurisé.

[0116] Le procédé est ainsi davantage sécurisé.

[0117] Comme déjà mentionné, le système 1, notamment la deuxième partie de l'implémentation en boite blanche, n'a pas besoin de vérifier que la donnée résultat correspond au résultat de l'application de la fonction intermédiaire à la donnée de traitement. Si la donnée résultat ne correspond pas au résultat de l'application de la fonction intermédiaire à la donnée de traitement, le message de sortie obtenu sera erroné. En effet, le message de sortie ne sera pas l'image du message d'entrée par la fonction sécurisée.

[0118] De préférence, le calcul du message de sortie à

partir de la donnée résultat pendant l'étape de calcul du message de sortie (étape E500), diffère d'une application à la donnée résultat, d'une fonction inverse de la fonction intermédiaire.

[0119] En outre, quand le procédé est selon le mode particulier de réalisation décrit ci-avant, le calcul (étape E500) du message de sortie à partir de la donnée résultat diffère de la fonction identité.

[0120] Le procédé est ainsi davantage sécurisé.

[0121] Quand le procédé est selon le mode particulier de réalisation et que le procédé comprend l'étape de calcul d'au moins une autre donnée de traitement (étape E110), ladite étape de calcul d'au moins une autre donnée de traitement (étape E110) est mise en oeuvre par la première partie de l'implémentation en boite blanche, et la deuxième partie de l'implémentation en boîte blanche utilise avantageusement l'au moins une autre donnée de traitement pour le calcul du message de sortie à partir de la donnée résultat (étape E500).

[0122] De façon également avantageuse, quand le procédé est selon l'autre mode particulier de réalisation décrit ci-avant et que le procédé comprend l'étape de calcul d'au moins une autre donnée de traitement (étape E110), le message de sortie comprend en outre l'autre donnée de traitement.

[0123] Le procédé est ainsi davantage sécurisé.

[0124] L'étape de calcul d'au moins une autre donnée de traitement (étape E110) contribue aux calculs de la fonction sécurisée en obtenant l'autre donnée de traitement. Le calcul de l'au moins une autre donnée de traitement, le cas échéant la deuxième autre fonction, est choisi de sorte à mettre en oeuvre la fonction sécurisée.

[0125] On notera que dans le procédé pour effectuer une fonction sécurisée selon l'invention, et/ou dans le système 1 selon l'invention, la fonction sécurisée peut se composer d'une suite d'opérations, la fonction intermédiaire étant une partie de ladite suite d'opérations.

[0126] Typiquement, la fonction sécurisée se compose d'une suite d'opérations mises en oeuvre sur des données d'entrée.

[0127] Une opération peut être une opération arithmétique ou une opération arithmétique booléenne, par exemples une addition, une soustraction, une multiplication, une division, une exponentiation, un logarithme.

[0128] Une opération peut également être une opération logique, par exemples la disjonction, la conjonction ou la négation.

[0129] Une donnée d'entrée peut comprendre tout ou partie du message d'entrée, et/ou tout ou partie d'au moins une donnée intermédiaire.

[0130] Une donnée intermédiaire est le résultat de l'application d'une opération de ladite suite d'opérations à au moins une donnée d'entrée.

[0131] Le calcul (étape E100) de la donnée de traitement à partir du message d'entrée, par l'implémentation en boite blanche, consiste alors en l'application d'une première partie de la suite d'opérations qui composent la fonction sécurisée.

[0132] La donnée de traitement est ainsi une donnée intermédiaire.

[0133] La première partie de la suite d'opérations qui composent la fonction sécurisée, comporte au moins une opération de ladite suite d'opérations qui composent la fonction sécurisée.

[0134] Typiquement, l'application d'une première autre fonction au message d'entrée ou à une première partie du message d'entrée consiste en la première partie de la suite d'opérations qui composent la fonction sécurisée.

[0135] L'étape d'obtention (étape E400) mise en oeuvre par l'élément sécurisé réalise alors une deuxième partie de la suite d'opérations qui composent la fonction sécurisée, ladite deuxième partie de la suite d'opérations qui composent la fonction sécurisée constituant la fonction intermédiaire appliquée à la donnée de traitement.

[0136] La deuxième partie de la suite d'opérations qui composent la fonction sécurisée, comporte au moins une opération de ladite suite d'opérations qui composent la fonction sécurisée.

[0137] La deuxième partie de la suite d'opérations qui composent la fonction sécurisée est typiquement disjointe de la première partie de la suite d'opérations décrite ci-avant.

[0138] La donnée de traitement est une donnée d'entrée pour la fonction intermédiaire.

[0139] La donnée résultat est le résultat de la mise en oeuvre de ladite deuxième partie de la suite d'opérations qui composent la fonction sécurisée. La donnée résultat est donc une donnée intermédiaire.

[0140] Quand le procédé est selon le mode particulier de réalisation décrit ci-avant, la donnée résultat est alors utilisée pour une donnée d'entrée d'au moins une opération d'une troisième partie de la suite d'opérations qui composent la fonction sécurisée.

[0141] Ladite au moins une opération d'une troisième partie de la suite d'opérations est mise en oeuvre dans le calcul (étape E500) du message de sortie à partir de la donnée résultat, par la deuxième partie de l'implémentation en boîte blanche.

[0142] La troisième partie de la suite d'opérations qui composent la fonction sécurisée est typiquement disjointe de la première partie de ladite suite d'opérations et de la deuxième partie de ladite suite d'opérations, décrites ci-avant.

[0143] Quand le procédé est selon l'autre mode particulier de réalisation décrit ci-avant, la donnée résultat peut être utilisée pour une donnée d'entrée d'au moins une opération d'une troisième partie de la suite d'opérations qui composent la fonction sécurisée. Ladite au moins une opération d'une troisième partie de la suite d'opérations est mise en oeuvre dans le calcul (étape E500) du message de sortie à partir de la donnée résultat, typiquement par le deuxième module du système 1.

[0144] On notera que l'étape de calcul (E110) d'au moins une autre donnée de traitement par l'implémenta-

tion en boîte blanche peut consister en la mise en oeuvre d'une quatrième partie de la suite d'opérations qui composent la fonction sécurisée.

[0145] Typiquement, l'application d'une deuxième autre fonction au message d'entrée, à une deuxième partie du message d'entrée, à la donnée de traitement ou à une première partie de la donnée de traitement consiste en ladite quatrième partie de la suite d'opérations qui composent la fonction sécurisée.

[0146] La quatrième partie de la suite d'opérations qui composent la fonction sécurisée, comporte au moins une opération de ladite suite d'opérations qui composent la fonction sécurisée.

[0147] La quatrième partie de la suite d'opérations qui composent la fonction sécurisée est typiquement disjointe de la première partie de ladite suite d'opérations, de la deuxième partie de ladite suite d'opérations, et de la troisième parties décrites ci-avant.

[0148] La figure 3 illustre un exemple de fonction sécurisée effectuée par un procédé ou un système selon l'invention.

[0149] La fonction sécurisée illustrée dans cette figure met en correspondance un message d'entrée I avec un message de sortie O, et permet l'exécution d'un algorithme de chiffrement symétrique E_k avec une clé cryptographique prédéterminée K, ladite exécution étant sécurisée contre les fautes en utilisant une contremesure infective.

[0150] La fonction sécurisée comprend l'obtention d'un premier résultat provisoire par une première exécution du chiffrement E_k appliqué au message d'entrée I, et l'obtention d'un deuxième résultat provisoire par une première exécution d'une fonction de diffusion D_R appliquée au premier résultat provisoire.

[0151] La fonction sécurisée comprend également l'obtention d'un troisième résultat provisoire par une deuxième exécution du chiffrement E_k appliqué au message d'entrée I, et l'obtention d'un quatrième résultat provisoire par une deuxième exécution de la fonction de diffusion D_R appliquée au troisième résultat provisoire.

[0152] La fonction sécurisée obtient alors le message de sortie O par une combinaison de type « ou exclusif » entre le deuxième résultat provisoire, le quatrième résultat provisoire et un autre résultat provisoire parmi le premier résultat provisoire et le troisième résultat provisoire.

[0153] La fonction de diffusion D_R est typiquement une fonction de hachage paramétrée avec une valeur R. La fonction sécurisée illustrée dans la figure 3 est décrite dans le document « A high-Order infective Countermeasure Framework », Guillaume Barbu ; Luk Bettale ; Laurent Castelnovi ; Thomas Chabrier ; Nicolas Débande ; Christophe Giraud ; Nathan Reboud, DOI : 10.1109/FDTC53659.2021.00012.

[0154] Un exemple de procédé selon l'invention, pour effectuer cette fonction sécurisée, consiste en ce que les deux exécutions du chiffrement E_k et une des exécutions de la fonction de diffusion D_R sont mises en oeuvre par l'implémentation en boîte blanche, typiquement pendant l'étape de calcul d'une donnée de traitement (étape E100) et l'étape de calcul d'au moins une autre donnée de traitement (étape E110).

[0155] L'autre donnée de traitement comprend le premier résultat provisoire ou le troisième résultat provisoire.

[0156] Si l'implémentation en boîte blanche met en oeuvre la première exécution de la fonction de diffusion D_R, la donnée de traitement comprend le troisième résultat provisoire et la valeur R, l'autre donnée de traitement comprend en outre le deuxième résultat provisoire, la fonction intermédiaire est la fonction de diffusion, et la donnée résultat est le quatrième résultat provisoire.

[0157] Si l'implémentation en boîte blanche met en oeuvre la deuxième exécution de la fonction de diffusion D_R, la donnée de traitement comprend le premier résultat provisoire et la valeur R, l'autre donnée de traitement comprend en outre le quatrième résultat provisoire, la fonction intermédiaire est la fonction de diffusion et la donnée résultat est le deuxième résultat provisoire.

[0158] Le calcul (étape E500) du message de sortie à partir de la donnée résultat est alors mis en oeuvre par la deuxième partie de l'implémentation en boite blanche, et obtient le message de sortie par la combinaison de type « ou exclusif » du deuxième résultat provisoire, du quatrième résultat provisoire et d'un autre résultat provisoire parmi le premier résultat provisoire et le troisième résultat provisoire, l'autre résultat provisoire étant le premier résultat provisoire, ou le troisième résultat provisoire, compris dans l'autre donnée de traitement.

[0159] Selon une variante de cet exemple, l'autre donnée de traitement a pour valeur la combinaison par « ou exclusif » d'un résultat provisoire parmi le premier résultat provisoire ou le troisième résultat provisoire, avec le deuxième résultat provisoire si l'implémentation en boîte blanche met en oeuvre la première exécution de la fonction de diffusion D_R, ou avec le quatrième résultat provisoire si l'implémentation en boîte blanche met en oeuvre la deuxième exécution de la fonction de diffusion D_R, et le calcul (étape E500) du message de sortie à partir de la donnée résultat obtient le message de sortie par la combinaison de type « ou exclusif » entre la donnée résultat et l'autre donnée de traitement.

[0160] Un homme du métier comprendra que les modes de réalisation, variantes, et différentes caractéristiques, décrits ci-avant peuvent être associées les unes avec les autres selon diverses combinaisons dans la mesure où elles ne sont pas incompatibles ou exclusives les unes des autres.

Revendications

1. Procédé pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie, le procédé étant mis en oeuvre par un système (1) comprenant un élément

sécurisé (3) et une implémentation en boîte blanche dans un environnement d'exécution non sécurisé (2), le procédé étant **caractérisé en ce qu'**il comprend les étapes suivantes :

- calcul (E100) d'une donnée de traitement à partir du message d'entrée par l'implémentation en boîte blanche,
- chiffrement (E200) de la donnée de traitement en utilisant une clé de chiffrement, par l'implémentation en boîte blanche,
- envoi (E300) de la donnée de traitement chiffrée à l'élément sécurisé,
- obtention (E400) par l'élément sécurisé d'une donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, la donnée résultat étant l'image de la donnée de traitement par une fonction intermédiaire différente de la fonction identité,
- calcul (E500) du message de sortie à partir de la donnée résultat.

2. Procédé pour effectuer une fonction sécurisée selon la revendication précédente dans lequel le message de sortie comprend la donnée résultat.

3. Procédé pour effectuer une fonction sécurisée selon l'une quelconque des revendications précédentes,

- le procédé comprenant en outre une étape de calcul (E110) d'au moins une autre donnée de traitement par l'implémentation en boîte blanche,
- et le message de sortie comprenant en outre l'autre donnée de traitement.

4. Procédé pour effectuer une fonction sécurisée selon la revendication 1 dans lequel :

- l'implémentation en boite blanche a une première partie et une deuxième partie ;
- le calcul (E100) d'une donnée de traitement à partir du message d'entrée et le chiffrement (E200) de la donnée de traitement en utilisant une clé de chiffrement, sont mis en oeuvre par la première partie de l'implémentation en boite blanche ;
- le calcul (E500) du message de sortie à partir de la donnée résultat est mis en oeuvre par la deuxième partie de l'implémentation en boite blanche ;

le procédé comprenant en outre une étape d'envoi de la donnée résultat à la deuxième partie de l'implémentation en boite blanche.

5. Procédé pour effectuer une fonction sécurisée selon la revendication précédente, le procédé comprenant en outre :

- le calcul (E110) d'au moins une autre donnée de traitement par la première partie de l'implémentation en boîte blanche, et
- l'utilisation de l'au moins une autre donnée de traitement par la deuxième partie de l'implémentation en boîte blanche pour le calcul du message de sortie à partir de la donnée résultat.

6. Procédé pour effectuer une fonction sécurisée selon l'une quelconque des revendications précédentes dans lequel :
l'obtention (E400) par l'élément sécurisé (3) de la donnée résultat, est par déchiffrement fonctionnel de la donnée de traitement chiffrée en utilisant une clé de déchiffrement fonctionnel pour la fonction intermédiaire, la clé de déchiffrement étant ladite clé de déchiffrement fonctionnel pour la fonction intermédiaire.

7. Procédé pour effectuer une fonction sécurisée selon la revendication précédente dans lequel la clé de chiffrement est une clé publique d'un algorithme RSA et la clé de déchiffrement fonctionnel est obtenue à partir d'une clé privée dudit algorithme RSA, la clé privé étant associée à la clé publique, l'algorithme RSA ayant un module de chiffrement déterminé et la fonction intermédiaire étant une exponentiation modulaire élevant la donnée de traitement à un exposant déterminé modulo le module de chiffrement déterminé.

8. Procédé pour effectuer une fonction sécurisée selon la revendication précédente dans lequel la clé de chiffrement comprend un exposant de chiffrement et la clé de déchiffrement fonctionnel comprend un inverse modulaire de l'exposant de chiffrement et l'exposant déterminé, ou le résultat du produit de l'exposant de chiffrement par l'exposant déterminé.

9. Procédé pour effectuer une fonction sécurisée selon l'une quelconque des revendications précédentes dans lequel :

- la clé de chiffrement n'intervient que dans l'étape (E200) de chiffrement de la donnée de traitement par l'implémentation en boîte blanche, et
- la clé de déchiffrement n'intervient que dans l'étape d'obtention (E400) d'une donnée résultat par l'élément sécurisé.

10. Procédé pour effectuer une fonction sécurisée selon l'une quelconque des revendications précédentes dans lequel la fonction sécurisée se compose d'une

suite d'opérations et la fonction intermédiaire est une partie de ladite suite d'opérations.

11. Procédé pour effectuer une fonction sécurisée selon l'une quelconque des revendications précédentes dans lequel la fonction sécurisée est une fonction cryptographique mettant en correspondance le message d'entrée avec le message de sortie à l'aide d'une clé cryptographique prédéterminée.

12. Procédé pour effectuer une fonction sécurisée selon la revendication précédente dans lequel la clé cryptographique prédéterminée est une clé distincte de la clé de chiffrement et de la clé de déchiffrement.

13. Système (1) pour effectuer une fonction sécurisée mettant en correspondance un message d'entrée avec un message de sortie, le système étant **caractérisé en ce qu'**il comprend :

- un premier composant comprenant tout ou partie d'une implémentation en boite blanche dans un environnement d'exécution non sécurisé (2), ladite toute ou partie de l'implémentation en boite blanche étant configurée pour calculer une donnée de traitement à partir du message d'entrée et chiffrer la donnée de traitement en utilisant une clé de chiffrement,
- un élément sécurisé (3) configuré pour recevoir la donnée de traitement chiffrée et obtenir une donnée résultat à partir de la donnée de traitement chiffrée et d'une clé de déchiffrement associée à la clé de chiffrement, la donnée résultat étant l'image de la donnée de traitement par une fonction intermédiaire différente de la fonction identité,
- un deuxième composant configuré pour recevoir la donnée résultat et calculer le message de sortie à partir de la donnée résultat.

14. Système (1) pour effectuer une fonction sécurisée selon la revendication précédente, dans lequel le message de sortie comprend la donnée résultat.

15. Système (1) pour effectuer une fonction sécurisée selon la revendication 13, dans lequel :

- l'implémentation en boite blanche a une première partie et une deuxième partie ;
- ladite toute ou partie de l'implémentation en boite blanche du premier composant est la première partie de l'implémentation en boite blanche ;
- le deuxième composant comprend la deuxième partie de l'implémentation en boite blanche dans l' environnement d'exécution non sécurisé (2), la deuxième partie de l'implémentation en boite blanche étant configurée pour recevoir la donnée résultat et calculer le message de sortie à partir de la donnée résultat.

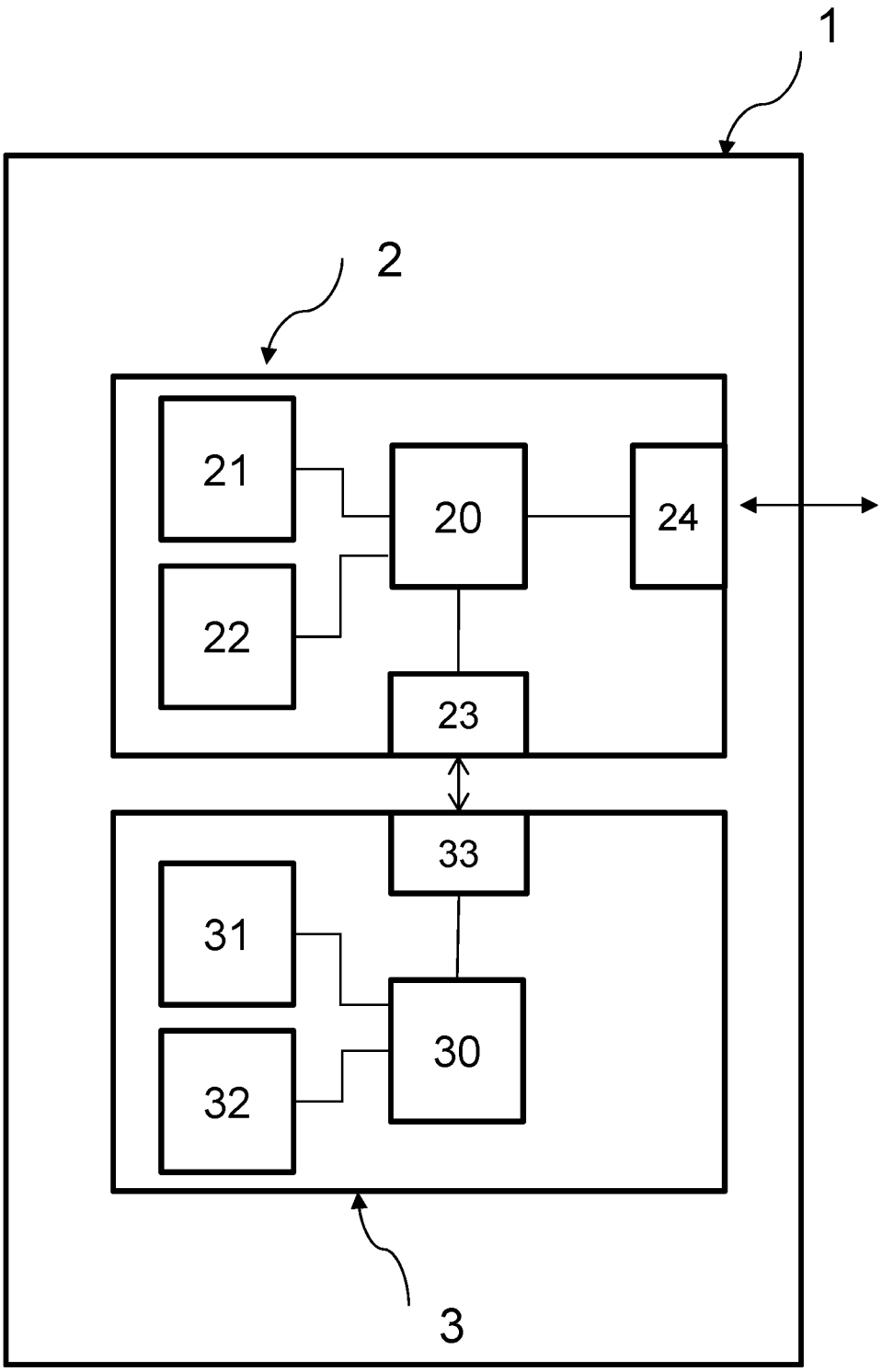

Figure 1

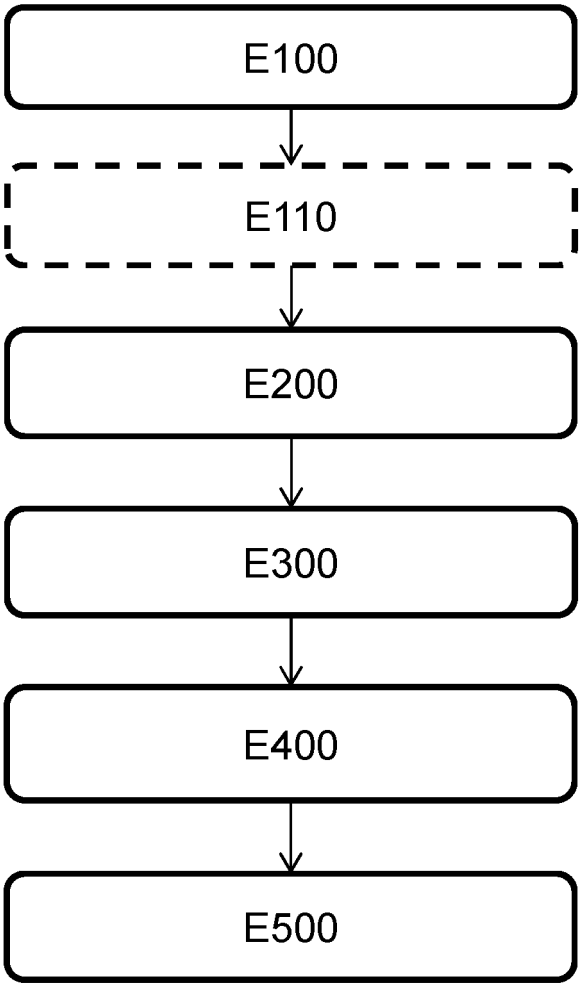

Figure 2

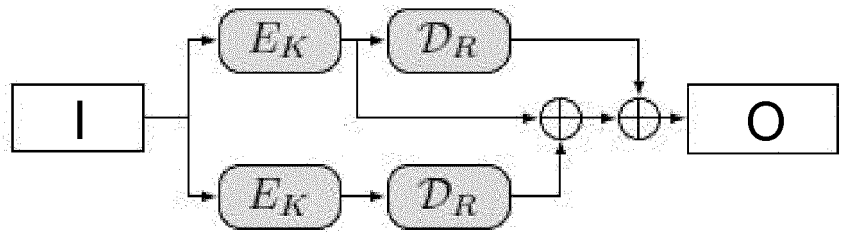

Figure 3

Europäisches Patentamt
European Patent Office
Office européen des brevets

RAPPORT DE RECHERCHE EUROPEENNE

Numéro de la demande
EP 24 20 9136

DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	Revendication concernée	CLASSEMENT DE LA DEMANDE (IPC)
X Y	EP 3 210 332 B1 (IRDETO BV [NL]) 29 janvier 2020 (2020-01-29) * alinéas [0071] - [0073], [0084] - [0088], [0097]; figure 2 *	1-5,9-15 4-8,15	INV. H04L9/00
X Y	US 2022/417001 A1 (SHAMSAASEF RAFIE [US] ET AL) 29 décembre 2022 (2022-12-29) * alinéas [0044] - [0046], [0051] - [0053]; figure 4 *	1-5,9-15 4-8,15	
Y	US 2016/182472 A1 (MICHIELS WIL [NL] ET AL) 23 juin 2016 (2016-06-23) * figures 1-4 *	4,5,15	
Y	BEN A FISCH ET AL: "Iron: Functional Encryption using Intel SGX", IACR, INTERNATIONAL ASSOCIATION FOR CRYPTOLOGIC RESEARCH, vol. 20170428:221846, 29 avril 2017 (2017-04-29), pages 1-37, XP061022762, DOI: 10.1145/3133956.3134106 * abrégé *	6-8	DOMAINES TECHNIQUES RECHERCHES (IPC) H04L

Le présent rapport a été établi pour toutes les revendications

Lieu de la recherche	Date d'achèvement de la recherche	Examinateur
Munich	7 janvier 2025	Billet, Olivier

CATEGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : arrière-plan technologique
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet antérieur, mais publié à la date de dépôt ou après cette date
D : cité dans la demande
L : cité pour d'autres raisons
.....
& : membre de la même famille, document correspondant

EPO FORM 1503 03.82 (P04C02)

ANNEXE AU RAPPORT DE RECHERCHE EUROPEENNE RELATIF A LA DEMANDE DE BREVET EUROPEEN NO. EP 24 20 9136

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche européenne visé ci-dessus.
Lesdits members sont contenus au fichier informatique de l'Office européen des brevets à la date du
Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets.

07-01-2025

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
EP 3210332 B1	29-01-2020	CA 2965032 A1	28-04-2016
		CN 107005402 A	01-08-2017
		EP 3210332 A1	30-08-2017
		US 2017237551 A1	17-08-2017
		WO 2016062609 A1	28-04-2016
US 2022417001 A1	29-12-2022	AR 126203 A1	27-09-2023
		CA 3222647 A1	29-12-2022
		GB 2622553 A	20-03-2024
		US 2022417001 A1	29-12-2022
		US 2024405975 A1	05-12-2024
		WO 2022271975 A1	29-12-2022
US 2016182472 A1	23-06-2016	CN 105718763 A	29-06-2016
		EP 3035582 A1	22-06-2016
		US 2016182472 A1	23-06-2016

EPO FORM P0460

Pour tout renseignement concernant cette annexe : voir Journal Officiel de l'Office européen des brevets, No.12/82

RÉFÉRENCES CITÉES DANS LA DESCRIPTION

Cette liste de références citées par le demandeur vise uniquement à aider le lecteur et ne fait pas partie du document de brevet européen. Même si le plus grand soin a été accordé à sa conception, des erreurs ou des omissions ne peuvent être exclues et l'OEB décline toute responsabilité à cet égard.

Documents brevets cités dans la description

- US 2019312718 A **[0009]**

Littérature non-brevet citée dans la description

- **S. CHOW et al.** White Box Cryptography and an AES implementation. *Post-Proceedings of the 9th Annual Workshop on Selected Areas in Cryptography (SAC'02)*, 2002, 15-16 **[0004]**
- **GUILLAUME BARBU** ; **LUK BETTALE** ; **LAURENT CASTELNOVI** ; **THOMAS CHABRIER** ; **NICOLAS DÉBANDE** ; **CHRISTOPHE GIRAUD** ; **NATHAN REBOUD**. *A high-Order infective Countermeasure Framework* **[0153]**